



# Level set topology optimization for design-dependent pressure loads using the reproducing kernel particle method

Andreas Neofytou<sup>1</sup> · Renato Picelli<sup>2</sup> · Tsung-Hui Huang<sup>3</sup> · Jiun-Shyan Chen<sup>3</sup> · H. Alicia Kim<sup>1,3</sup>

Received: 10 June 2019 / Revised: 11 December 2019 / Accepted: 13 February 2020  
© The Author(s) 2020

## Abstract

This paper presents a level set topology optimization method in combination with the reproducing kernel particle method (RKPM) for the design of structures subjected to design-dependent pressure loads. RKPM allows for arbitrary particle placement in discretization and approximation of unknowns. This attractive property in combination with the implicit boundary representation given by the level set method provides an effective framework to handle the design-dependent loads by moving the particles on the pressure boundary without the need of remeshing or special numerical treatments. Moreover, the reproducing kernel (RK) smooth approximation allows for the Young's modulus to be interpolated using the RK shape functions. This is another advantage of the proposed method as it leads to a smooth Young's modulus distribution for smooth boundary sensitivity calculation which yields a better convergence. Numerical results show good agreement with those in the literature.

**Keywords** Topology optimization · Level set method · Design-dependent pressure load · Reproducing kernel particle method

## 1 Introduction

Design-dependent loading of interest in this paper is the surface loads such as pressure acting on the boundaries of the structure. When the boundaries themselves are subject to a change by optimization, this presents a challenging class of problems in topology optimization. Such design-dependent surface or boundary loads can be found in many engineering

structures. Pressure vessels, civil structures subjected to wind and snow loading, and underwater structures subjected to external fluid pressure are typical examples. The main challenge in topology optimization with design-dependent loads lies in determining the surface on which the load will act.

In density-based methods, the presence of elements with intermediate densities near the boundaries makes it difficult to identify the specific surface for the moving loads to be applied. Hammer and Olhoff (2000) proposed the use of isodensity nodal points and Bezier spline curves. The volumetric density of material is used to define the load surface which is represented by spline functions with control points that depend on the design variables. The pressure acting on this surface is then transformed into nodal forces in the finite element analysis (FEA) model. A main issue with this methodology, however, is the appearance of ill-defined loading curves if a cutoff value of the material density is not appropriately selected. As a consequence, invalid load surfaces may appear. To overcome this, Du and Olhoff (2004) improved this method by determining the loading surface in a given step based on the isoline of material density not only in the current step but also in the previous one. Furthermore, the loading surface is represented directly by straight segments instead of a spline curve, thus making the calculation of nodal forces simpler. The same framework for tracking the moving load surfaces was later adopted by Lee and Martins (2012).

---

Responsible Editor: Fred van Keulen

✉ H. Alicia Kim  
alicia@ucsd.edu

Andreas Neofytou  
NeofytouA@cardi.ac.uk

Renato Picelli  
rpicelli@usp.br

<sup>1</sup> Cardiff School of Engineering, Cardiff University, The Queen's Buildings, 14-17 The Parade, Cardiff, SP CF24 3AA, UK

<sup>2</sup> Department of Mining and Petroleum Engineering, University of São Paulo, Praça Narciso de Andrade Vila Mathias, Santos, SP 11013-560, Brazil

<sup>3</sup> Structural Engineering Department, University of California, San Diego, 9500 Gilman Drive, San Diego, CA 92093, USA

Another way of tracking pressure load surfaces in solid isotropic material with penalization (SIMP) was introduced by Chen and Kikuchi (2001) and also used by Bourdin and Chambolle (2003). Instead of constructing a parameterized surface for the pressure to act on, the design-dependent loads are simulated by fictitious thermal loads. A fluid domain is introduced along with the solid and void domains. This results in a three-phase material distribution problem within the design domain in which the solid, void, and hydrostatic fluid phases are optimally distributed. A hydrostatic pressure force exists at the interface between the solid and fluid regions and it is simulated by the thermal load due to the mismatch of thermal expansion coefficients of the two materials. The thermal stress tensor of the non-fluid area is set to be constant regardless of the density distribution to ensure that pressure does not act between the solid and fluid regions. Later, Sigmund and Clausen (2007) introduced a way to solve the pressure load problem based on a mixed displacement-pressure FE formulation. Pressure is included as a separate variable and is used to define the void phase as hydrostatic incompressible fluid. The pressure load is transferred in the domain through the incompressible hydrostatic fluid.

Bidirectional evolutionary structural optimization (BESO) does not employ the continuous density variable, thus avoiding some of the challenges in SIMP. Picelli et al. (2015) proposed a BESO approach where binary solid-void design variables are used along with the process of fluid flooding which allows the fluid and structure to be modeled during optimization with separate domains. Sivapuram and Picelli (2017) recently created the topology optimization of binary structure (TOBS) method and applied a similar approach to solve for a design-dependent fluid pressure problem. However, the piecewise constant discrete nature of BESO often yield the boundaries to be represented by finite elements' jagged edges.

In contrast, the level set topology optimization method has an advantage that the boundary can be clearly represented as the structure is implicitly described by a level set function. Therefore, solid and void regions are well defined. However, without remeshing, there are no nodes along the pressure boundaries to apply the loading. In the seminal paper by Allaire et al. (2004), a Dirac delta function was used together with the ersatz material approximation to replace the surface loads by equivalent volume forces, thus avoiding the need for a pressure surface identification. Using a similar approach, Xia et al. (2015) employed two level set functions to represent the free and pressure boundaries. Shu et al. (2014) also used a Dirac delta function to minimize sound pressure in an acoustic-structural system. More recently, Emmendoerfer et al. (2018) proposed a level set topology optimization in which the surface loads acting on the moving boundary are transformed into work-equivalent nodal forces to solve pressure loading problems with level set topology optimization and a fixed grid. In this work, the ersatz material

was also used for the elements cut by the boundary and a virtual fluid flooding process was used to track the pressure surface. A similar approach has recently been employed by Picelli et al. (2019) where a discretized fluid domain was used to explore benchmarking examples with purely hydrostatic pressure loading.

Alternatively to the ersatz material approximation, other approaches have been used in combination with the level set topology optimization method such as the extended finite element method (XFEM) or remeshing the geometry at every iteration. To model the structural boundary using XFEM, the state variable interpolation is enriched to account for discontinuities of the variables within an element that is cut by an interface. Local remeshing is then performed within the intersected elements for the integration of the weak form. Jenkins and Maute (2016) combined a level set method XFEM to track fluid-structure interface in a fluid-structure interaction problem. A remeshing algorithm was used by Isakari et al. (2017) to track the acoustic interfaces in a FE analysis coupled with the boundary element (BE) method.

A new class of analysis methods has emerged in the last two decades, in which no mesh is needed and shape functions are constructed from a set of points. Meshfree methods are designed to inherit the useful characteristics of the FEM, such as compact supports of shape functions and good approximation properties, and at the same time overcome the main disadvantages caused by the mesh dependence. Over the years, a wide variety of meshfree methods have been studied and they can generally be categorized into collocation or Galerkin meshfree methods. Collocation methods are based on the strong form of partial differential equations (PDEs). Galerkin meshfree methods on the other hand are based on the weak form of PDEs. The element free Galerkin method (EFG) (Belytschko et al. 1994) and the reproducing kernel particle method (RKPM) (Liu et al. 1995; Chen et al. 1996) are typical Galerkin meshfree methods. A recent review on meshfree methods can be found in Chen et al. (2017b). Meshfree methods have been used to study a wide variety of problems in engineering as an alternative to the FEM. Some examples include fracture mechanics (Belytschko et al. 1994), large deformation problems (Chen et al. 1997), and contact mechanics (Wang et al. 2014).

Meshfree methods have been employed in topology optimization. Both RKPM and EFG were used with SIMP to optimize geometrically non-linear (Cho and Kwak 2006; Du et al. 2009; Zhang et al. 2018) and elastostatic structures (Zhou and Zou 2008). Kim et al. (2003) used RKPM for shape optimization of 2D and 3D elastic structures subjected to fixed point loads. Luo et al. (2012) combined the parametric level set topology optimization method with a meshfree analysis approach based on the compactly supported basis functions (CSRBFs). CSRBFs are used in this work for both parameterizing the level set function and construct the shape functions for meshfree approximations. A background mesh is used for

domain integration. The particles were placed on the nodal positions of the background mesh and their positions remained constant throughout the optimization process. To account for the boundary discontinuity, the CSRBFs were used to interpolate the level set function at quadrature points based on the discrete level set function nodal values. Quadrature points with a positive or zero level set value were considered as solid, whereas a weak material was assigned to points with negative level set values. The proposed meshfree level set method was used to solve compliance minimization for linear elasticity under constant point loads (Luo et al. 2012). Recently, Khan et al. (2018) employed the level set topology optimization method in combination with the element-free Galerkin method. In this work, compliance minimization of 2D linear elastic problems subjected to fixed point loads is considered. Similarly to Luo et al. (2012), a background mesh is used for the integration with particle positions being fixed at nodal positions of the background mesh during optimization.

This paper presents a numerical scheme for level set topology optimization for design-dependent pressure loads with meshfree RKPM. Inserting new boundary particles for the pressure loading, which can be done straightforwardly by RKPM (Liu et al. 1997; You et al. 2003), is an advantage of using this method. This eliminates the need to transform the loads or re-mesh the structure. The clear boundary representation provided by the level set method in combination with the ability to freely place RKPM particles wherever needed in the design domain allows for a clear identification of the pressure boundary. To identify the load carrying portion boundary and separate it from the free boundary, the process of fluid flooding as first called by Chen and Kikuchi (2001) is used.

The remainder of the paper is organized as follows: The “[Level set topology optimization method](#)” section outlines the level set topology optimization method. The “[Reproducing kernel particle method](#)” section describes RKPM and the construction of the reproducing kernel shape functions. The governing equations, imposition of boundary conditions, and domain integration are also discussed. In the “[Topology optimization with RKPM](#)” section, the numerical scheme for applying RKPM and the level set topology optimization is discussed, followed by the numerical investigations against the benchmarking examples in the “[Numerical examples](#)” section. The “[Conclusions](#)” section offers some concluding remarks.

## 2 Level set topology optimization method

This section briefly summarizes the level set topology optimization method used in this study. More details of the method can be found in Hedges et al. (2017) and Picelli et al. (2018).

In the level set topology optimization, the structural boundary is defined as the zero level set of an implicit function:

$$\begin{cases} \phi(\mathbf{x}) \geq 0 & \mathbf{x} \in \Omega \\ \phi(\mathbf{x}) = 0 & \mathbf{x} \in \Gamma \\ \phi(\mathbf{x}) < 0 & \mathbf{x} \notin \Omega \end{cases} \quad (1)$$

where  $\phi$  is the level set function,  $\Omega$  is the structural domain, and  $\Gamma$  is the structural boundary. Conventionally, the implicit function is initialized as a signed distance function (Wang et al. 2003; Allaire et al. 2004).

The structural boundary is optimized by iteratively solving the following Hamilton-Jacobi equation

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + |\nabla \phi(\mathbf{x})| V_n(\mathbf{x}) = 0 \quad (2)$$

where  $t$  is a fictitious time domain for the level set evolution and  $V_n$  is the normal velocity.

The level set function at each node is updated by solving the following discretized Hamilton-Jacobi equation using upwind differential scheme,

$$\phi_r^{k+1} = \phi_r^k - \Delta t |\nabla \phi_r^k| V_{nr} \quad (3)$$

where  $r$  is a discrete node in the design domain,  $V_{nr}$  is the normal velocity at node  $r$ ,  $k$  is the iteration number, and  $|\nabla \phi_r^k|$  is computed for each node using the Hamilton-Jacobi weighted essentially non-oscillatory method (HJ-WENO). To improve the computational efficiency, the level set update is restricted to nodes within a narrow band close to the boundary. This means that  $\phi_r$  is given by a signed distance to the boundary only within the narrow band. To correct for this effect,  $\phi_r$  is periodically reinitialized to a signed distance function. For the reinitialization and velocity extension, the fast marching method is used (Sethian 1996).

The velocities required for the level set update are obtained by solving the linearized optimization problem,

$$\begin{aligned} & \text{minimize} \quad \frac{\partial f}{\partial \Omega^k} \Delta \Omega^k \\ & \text{subject to} \quad \frac{\partial g_m}{\partial \Omega^k} \Delta \Omega^k \leq -g_m^{-k} \end{aligned} \quad (4)$$

where  $f$  is the objective function,  $g_m$  is the  $m^{\text{th}}$  inequality constraint function,  $\Delta \Omega^k$  is the update for the design domain  $\Omega$ , and  $g_m^{-k}$  is the change in the  $m^{\text{th}}$  constraint. Shape derivatives that provide information about how the objective and constraint functions change with respect to the boundary

movement typically take the form of boundary integrals (Allaire et al. 2004):

$$\frac{\partial f}{\partial \Omega} \Delta \Omega = \Delta t \int_{\Gamma} s_f V_n d\Gamma \quad (5)$$

$$\frac{\partial g_m}{\partial \Omega} \Delta \Omega = \Delta t \int_{\Gamma} s_{gm} V_n d\Gamma \quad (6)$$

where  $s_f$  and  $s_{gm}$  are the shape sensitivity functions for the objective and the  $m^{th}$  constraint, respectively. The integrals in Eqs. (5) and (6) can be estimated as,

$$\frac{\partial f}{\partial \Omega} \Delta \Omega \approx \sum_{j=1}^{nb} \Delta t V_{nj} s_{f,j} l_j = \mathbf{C}_f \cdot \mathbf{V}_n \Delta t \quad (7)$$

$$\frac{\partial g_m}{\partial \Omega} \Delta \Omega \approx \sum_{j=1}^{nb} \Delta t V_{nj} s_{gm,j} l_j = \mathbf{C}_{gm} \cdot \mathbf{V}_n \Delta t \quad (8)$$

where  $j$  is a discrete boundary point and  $V_{nj}$ ,  $s_{f,j}$ , and  $s_{gm,j}$  are the normal velocity and sensitivities for the objective and  $m^{th}$  constraint functions, respectively, at point  $j$ .  $l_j$  is the length of the local boundary around the boundary point  $j$ ,  $\mathbf{C}_f$  and  $\mathbf{C}_{gm}$  are vectors containing the product of boundary lengths and shape sensitivities and  $\mathbf{V}_n$  is the vector of normal velocities.

For a constrained problem, the following can be written,

$$\mathbf{V}_n \Delta t = \alpha \mathbf{d} \quad (9)$$

where  $\mathbf{d}$  is the search direction for the boundary update and  $\alpha > 0$  is the distance travelled along the search direction. To obtain the optimal boundary velocities, the optimization problem in Eq. (4) can be reformulated as

$$\begin{aligned} & \underset{\alpha^k, \lambda^k}{\text{minimize}} && \Delta t \mathbf{C}_f^k \cdot \mathbf{V}_n^k(\alpha^k, \lambda^k) \\ & \text{subject to} && \Delta t \mathbf{C}_m^k \cdot \mathbf{V}_n^k(\alpha^k, \lambda^k) \leq -\mathbf{g}_m^k \\ & && \mathbf{V}_{n,\min}^k \leq \mathbf{V}_n^k \leq \mathbf{V}_{n,\max}^k \end{aligned} \quad (10)$$

where  $\lambda$  are Lagrange multipliers for each constraint function. Equation (10) is solved for  $\alpha^k$  and  $\lambda^k$  at every iteration  $k$ . This method is implemented in the object-oriented C++ code (Sandilya et al. 2018) and is available as open source at <http://m2do.ucsd.edu/software/>.

### 3 Reproducing kernel particle method

This section outlines the reproducing kernel particle method (RKPM). For more details, the readers are referred to Liu et al. (1995) and Chen et al. (1996).

#### 3.1 Reproducing kernel approximation

To construct the RK approximation for a finite dimensional solution of the PDEs, the domain  $\Omega$  is discretized by a set of nodes  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{NP}\}$ , where  $\mathbf{x}_I$  is the position vector of node  $I$ , and  $NP$  is the total number of particles, then the RK approximation of a function  $u(\mathbf{x})$ , denoted by  $u^h(\mathbf{x})$ , is expressed as,

$$u(\mathbf{x}) \approx u^h(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x}) u_I \quad (11)$$

where  $\Psi_I(\mathbf{x})$  is the RK shape function at node  $I$ , and  $u_I$  is the corresponding nodal coefficients to be determined. Then the RK shape function can be expressed as

$$\Psi_I(\mathbf{x}) = C(\mathbf{x}; \mathbf{x} - \mathbf{x}_I) \Phi_a(\mathbf{x} - \mathbf{x}_I) \quad (12)$$

where  $\Phi_a(\mathbf{x} - \mathbf{x}_I)$  is the kernel function centered at  $\mathbf{x}_I$  with support size  $a$ . The kernel function controls the smoothness and locality of the approximation function, and hence, it should be chosen according to the characteristics of the problem such as the order and type of the PDEs. It is common to use cubic B-spline function as the kernel function,

$$\Phi_a(\mathbf{x} - \mathbf{x}_I) = \begin{cases} \frac{2}{3} - 4z_I^2 + 4z_I^3 \text{ for } & z_I \leq \frac{1}{2} \\ \frac{4}{3} - 4z_I + 4z_I^2 - \frac{4}{3}z_I^3 \text{ for } & \frac{1}{2} < z_I \leq 1 \\ 0 \text{ for } & z_I > 1 \end{cases} \quad (13)$$

where  $z_I$  is defined as:

$$z_I = \|\mathbf{x} - \mathbf{x}_I\| / a_I \quad (14)$$

To ensure smoothness in our shape sensitivities, which involves computing stress and strain terms at the boundary points, we chose here the cubic B-spline kernel function with  $C^2$  continuity. The higher order smoothness achieved by the RK shape functions using the cubic B-spline kernel has been demonstrated in many examples in the literature of RKPM. For example, it has been shown that a smooth transition in material properties can be achieved by the smooth RK approximation in the modelling of biomaterials (Chen et al. 2017a). Other examples can be found in the context of contact mechanics (Wang et al. 2014) and large deformation analysis of non-linear structures (Chen et al. 1996). Other kernel function with different level of smoothness and locality can be found in Huang et al. (2019), and the numerical comparison of using different kernel functions can be found in Belytschko et al. (1994).

The correction function  $C(\mathbf{x}; \mathbf{x} - \mathbf{x}_I)$  is introduced to ensure the completeness of the RK approximation by enforcing the following  $n$ th order reproducing conditions



$$C(\mathbf{x}; \mathbf{x}-\mathbf{x}_I) = \sum_{|\beta|=0}^n (\mathbf{x}-\mathbf{x}_I)^\beta b_\beta(\mathbf{x}) = \mathbf{H}^T(\mathbf{x}-\mathbf{x}_I) \mathbf{b}(\mathbf{x}) \quad (15)$$

where  $n$  is the specified order of completeness, which is related to the order of consistency in solving PDEs,  $\beta = (\beta_1, \beta_2, \beta_3)$  is the three-dimensional index, and  $|\beta| \equiv \sum_{i=1}^3 \beta_i$ .  $\mathbf{e} b_\beta(\mathbf{x})$  is the corresponding coefficient of the monomials  $(\mathbf{x}-\mathbf{x}_I)^\beta$ , where  $(\mathbf{x}-\mathbf{x}_I)^\beta \equiv (x_1-x_{1I})^{\beta_1} (x_2-x_{2I})^{\beta_2} (x_3-x_{3I})^{\beta_3}$  and  $\mathbf{H}^T(\mathbf{x}-\mathbf{x}_I)$  can be expressed as

$$\mathbf{H}^T(\mathbf{x}-\mathbf{x}_I) = \left[ 1, \quad x-x_I, \quad y-y_I, \quad z-z_I, \quad (x-x_I)^2, \quad \dots, \quad (z-z_I)^n \right] \quad (16)$$

and  $\mathbf{b}^T(\mathbf{x})$  is the unknown coefficient vector which can be computed by the  $n$ th order reproducing conditions

$$\sum_{I=1}^{NP} \Psi_I(\mathbf{x}) \mathbf{x}_I^\beta = \mathbf{x}^\beta, \quad |\beta| \leq n \quad (17)$$

and by substituting Eqs. (12) and (15) into Eq. (17). The coefficients  $\mathbf{b}(\mathbf{x})$  can be determined as

$$\mathbf{b}(\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{x}) \mathbf{H}(\mathbf{0}) \quad (18)$$

where  $\mathbf{M}(\mathbf{x})$  is the moment matrix. The support size  $a_I$  is typically defined as:

$$a_I = ch_I \quad (19)$$

where  $c$  is the normalized support size, chosen between 1.5 and 2.0 in practice, and  $h_I$  is the nodal spacing associated with point  $\mathbf{x}_I$  (Huang et al. 2019). This choice for the support has been shown to be stable in the literature (Belytschko et al. 1994; Chen et al. 1996; Liu et al. 1997).

The substitution of Eq. (18) into Eq. (15) results in

$$\mathbf{M}(\mathbf{x}) = \sum_{I=1}^{NP} \mathbf{H}(\mathbf{x}-\mathbf{x}_I) \mathbf{H}^T(\mathbf{x}-\mathbf{x}_I) \Phi_a(\mathbf{x}-\mathbf{x}_I) \quad (20)$$

where  $\mathbf{H}^T(\mathbf{x}-\mathbf{x}_I)$  is the vector of monomial basis functions. Through the combination of Eqs. (18), (15), and (12), the RK shape function in Eq. (12) can then be obtained:

$$\Psi_I(\mathbf{x}) = \mathbf{H}^T(\mathbf{0}) \mathbf{M}^{-1}(\mathbf{x}) \mathbf{H}(\mathbf{x}-\mathbf{x}_I) \Phi_a(\mathbf{x}-\mathbf{x}_I) \quad (21)$$

### 3.2 Galerkin approximation and formulation

The RK shape functions lack the Kronecker delta property. For this reason, applying the essential boundary conditions is not straightforward. Several approaches have been proposed for this purpose. In this paper, the Lagrange multiplier method is used to enforce the essential boundary conditions

because of its numerical simplicity and good accuracy (Belytschko et al. 1994). Using the Lagrange multiplier method to impose the essential boundary conditions, the discrete equations corresponding to the weak formulation of 2-dimensional linear elasticity can be expressed by

$$\begin{bmatrix} \mathbf{K} & \mathbf{G}^T \\ \mathbf{G} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{q} \end{bmatrix} \quad (22)$$

where

$$\mathbf{K}_{IJ} = \int_{\Omega} \mathbf{B}_I^T \mathbf{D} \mathbf{B}_J d\Omega \quad (23)$$

$$\mathbf{G}_{IJ} = - \int_{\Gamma^{g_i}} \Psi_I \mathbf{N}_J d\Gamma \quad (24)$$

$$\mathbf{f}_I = \int_{\Omega} \Psi_I \mathbf{b} d\Omega + \int_{\Gamma^{h_i}} \Psi_I \mathbf{h} d\Gamma \quad (25)$$

$$\mathbf{q}_I = - \int_{\Gamma^{g_i}} \mathbf{N}_I^T \mathbf{g} d\Gamma \quad (26)$$

$$\Psi_I = \begin{bmatrix} \Psi_I & 0 \\ 0 & \Psi_I \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_x \\ b_y \end{bmatrix} \quad \mathbf{h} = \begin{bmatrix} h_x \\ h_y \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} g_x \\ g_y \end{bmatrix} \quad (27)$$

$$\mathbf{B}_I = \begin{bmatrix} \Psi_{I,x} & 0 \\ 0 & \Psi_{I,y} \\ \Psi_{I,y} & \Psi_{I,x} \end{bmatrix} \quad \mathbf{N}_I = \begin{bmatrix} N_I & 0 \\ 0 & N_I \end{bmatrix} \quad (28)$$

$$\mathbf{D} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix} \quad (29)$$

In the above equations,  $\mathbf{K}$  is the system stiffness matrix;  $\mathbf{u}$  is the vector of structural displacements;  $\mathbf{G}$  is a matrix to enforce boundary conditions using the Lagrange multiplier method;  $\mathbf{f}$  represents the external load due to the body force,  $\mathbf{b}$ , and surface load,  $\mathbf{h}$ ; and  $\mathbf{B}$  is the geometric strain-displacement matrix. Matrix  $\mathbf{D}$  is the matrix of material elastic constants with  $E$ , the Young's modulus, and Poisson's ratio  $\nu$ .  $\Gamma^{h_i}$  is the Neumann boundary and  $\Gamma^{g_i}$  is the Dirichlet boundary with a prescribed displacement,  $\mathbf{g}$ , to be enforced.  $N_I$  in Eqs. (26) and (28) is the standard Lagrangian interpolant along the boundary to be enforced.

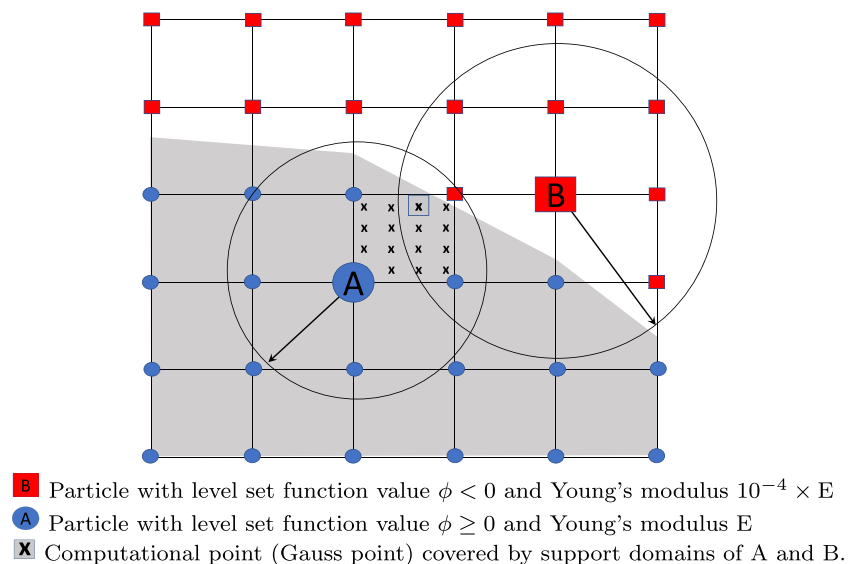
### 3.3 Domain integration

Since a Galerkin meshless method is used, domain integration is required. In this work, Gauss quadrature on a fixed rectangular background mesh is employed. As has been shown by Dolbow and Belytschko (1999), when Gauss quadrature is used for the domain integration, significant errors may arise when the background mesh do not coincide with the support domains. This becomes particularly difficult when circular supports are being used. As has been shown by Dolbow and Belytschko (1999), this can be improved by using sufficiently high quadrature rules. Following this work, each background

cell has  $4 \times 4$  quadrature points in our implementation to ensure accuracy. Integration schemes based on nodal integration rather than Gauss integration have also been proposed in the literature such as the stabilized conforming nodal integration (SCNI) proposed by Chen et al. (2001). Investigation of different integration schemes and studying the effects of these schemes on optimization is a topic for future work. It is important to note that RKPM is not limited to a regular background mesh. Furthermore, the positions of the particles within the design domain are independent from the background mesh. This very attractive feature is one of the main advantages exploited in this work. Figure 1 illustrates how the moving boundary discontinuity is represented. Level set function defines whether a point lies inside or outside the structure, and Young's modulus values for solid and weak materials are also known. Furthermore, an interpolation of the Young's modulus at quadrature points is performed using the RK shape functions to achieve a smooth Young's modulus distribution. This in turn leads to a smooth boundary sensitivity field for a better convergence. Gauss points that lie near the structural boundary are covered by the support domains of particles in both the solid and void regions as shown in Fig. 1. At these points, Young's modulus  $E$  is computed by RK approximation using the level set function values at the particles whose RK support domains cover the Gauss point. Particles with  $\phi \geq 0$  (for example, particle A in Fig. 1) have a Young's modulus value  $E$ , whereas particles with  $\phi < 0$  (for example particle B in Fig. 1) have a Young's modulus value equal to  $10^{-4}E$ ,

$$E \text{ at node } I : \begin{cases} E(x_I) = E \text{ if } \phi \geq 0 \\ E(x_I) = 10^{-4} \times E \text{ if } \phi < 0 \end{cases} \quad (30)$$

**Fig. 1** Interpolating Young's modulus at computational point (Gauss point)



where  $E(x_I)$  is the Young's modulus associated with the  $I^{\text{th}}$  node and  $E$  at Gauss point  $x_{\text{gp}}$  is computed as

$$E(x_{\text{gp}}) = \sum_{I \in G_{\text{gp}}} \psi_I(x_{\text{gp}}) E_I \quad (31)$$

where  $G_{\text{gp}}$  is the node set containing all nodes with their support covering the evaluation point  $x_{\text{gp}}$  and  $\psi_I$  represents the RK shape function. The obtained  $E$  is then used to compute the  $D$  matrix in Eq. (29).

## 4 Topology optimization with RKPM

### 4.1 Compliance minimization problem

Topology optimization in this study considers the well-known compliance minimization problem

$$\begin{aligned} \min \quad & J = l(u) \\ \text{s.t.} \quad & a(u, v) = l(v), \quad \forall v \in U \end{aligned} \quad (32)$$

$$V_s(\Omega) - V \leq 0$$

where the energy bilinear functional  $a(u, v)$  and the load linear form  $l(v)$  are defined as:

$$a(u, v) = \int_{\Omega} \{\epsilon(u)\}^T : D : \{\epsilon(v)\} d\Omega \quad (33)$$

$$l(v) = \int_{\Gamma_N} p \cdot v d\Gamma \quad (34)$$

Here,  $U$  is the space of kinematically admissible displacement fields.  $V_s(\Omega)$  is the volume fraction of the structure with respect to the design domain,  $V$  is the maximum allowed

volume fraction,  $\epsilon$  is the strain tensor,  $v$  is the virtual displacement,  $\Gamma_N$  is the Neumann boundary on which the pressure load is applied, and  $p$  is the pressure load. The pressure load is assumed to be constant although the method can be generalized for varying pressure.

$$p = -p_0 n \quad (35)$$

where  $p_0$  is the constant magnitude of the pressure load and  $n$  is the surface normal.

## 4.2 Particle placement

At every iteration, particles are added on the boundary at the intersection points between the background mesh and the boundary. Boundary particles from the previous iteration are removed. This means that particles do not move to different locations between iterations but instead they are generated. It is thus only required to compute the intersection points and not to track the particles. The process of generating new particles on the boundary is shown in Fig. 2. Initially, domain particles are placed at the nodal positions of the background mesh. These particles are shown with blue circles in Fig. 2a. At every iteration during the optimization process, these domain particles are generated first. As the boundary moves, domain particles are separated into particles inside the structure shown in blue circles in Fig. 2b and particles outside the structure shown in red circles. Particles outside the structure are kept to be used in the domain integration scheme with a fixed background mesh described in the “Reproducing kernel particle method” section. At every iteration, particles are added to the structural boundary. In Fig. 2b, these boundary particles are indicated by blue circles at the points where the updated boundary crosses the edges of the background mesh. Finally, if it happens that a domain particle is near to a boundary particle, the domain particle is removed as shown by the yellow “x” symbols. In the numerical tests, the removal does not influence the final topology but it produces a faster convergence. In fact, even if these particles were not removed at all, the final topologies do not change. The minimum distance

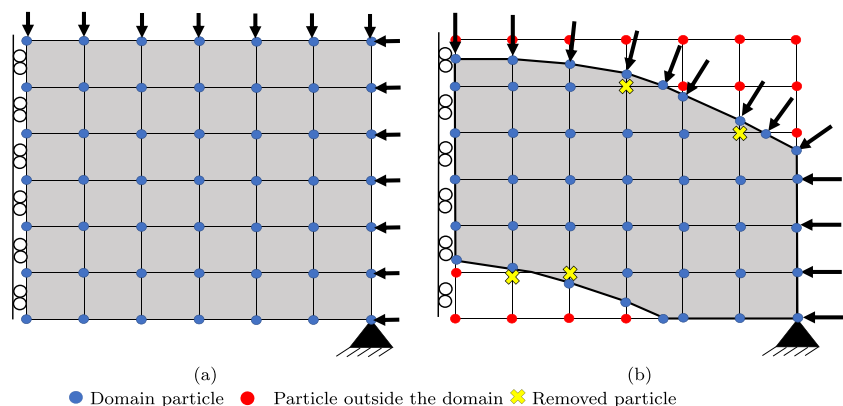
between particles was set to 0.05 times the length of a background element.

To identify the part of the boundary on which the pressure load is to be applied, a process similar to the fluid flooding proposed by Chen and Kikuchi (2001) is used here. The difference in this work is that instead of elements, we use the particles to identify the loading surface. Before optimization starts, the particles on the boundary segments carrying the initial pressure loads are marked as “pressure” particles. Depending on the signed distance value at the particles and the position of the pressure boundary, the following particle types emerge in addition to pressure particles:

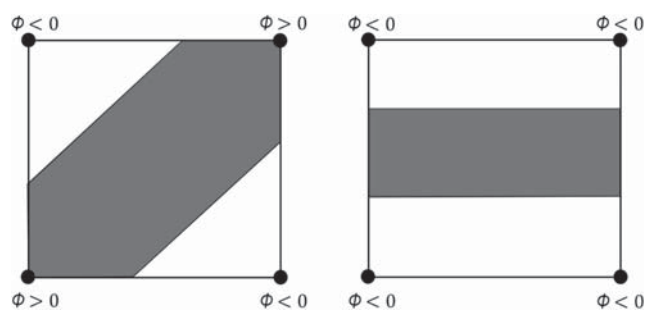
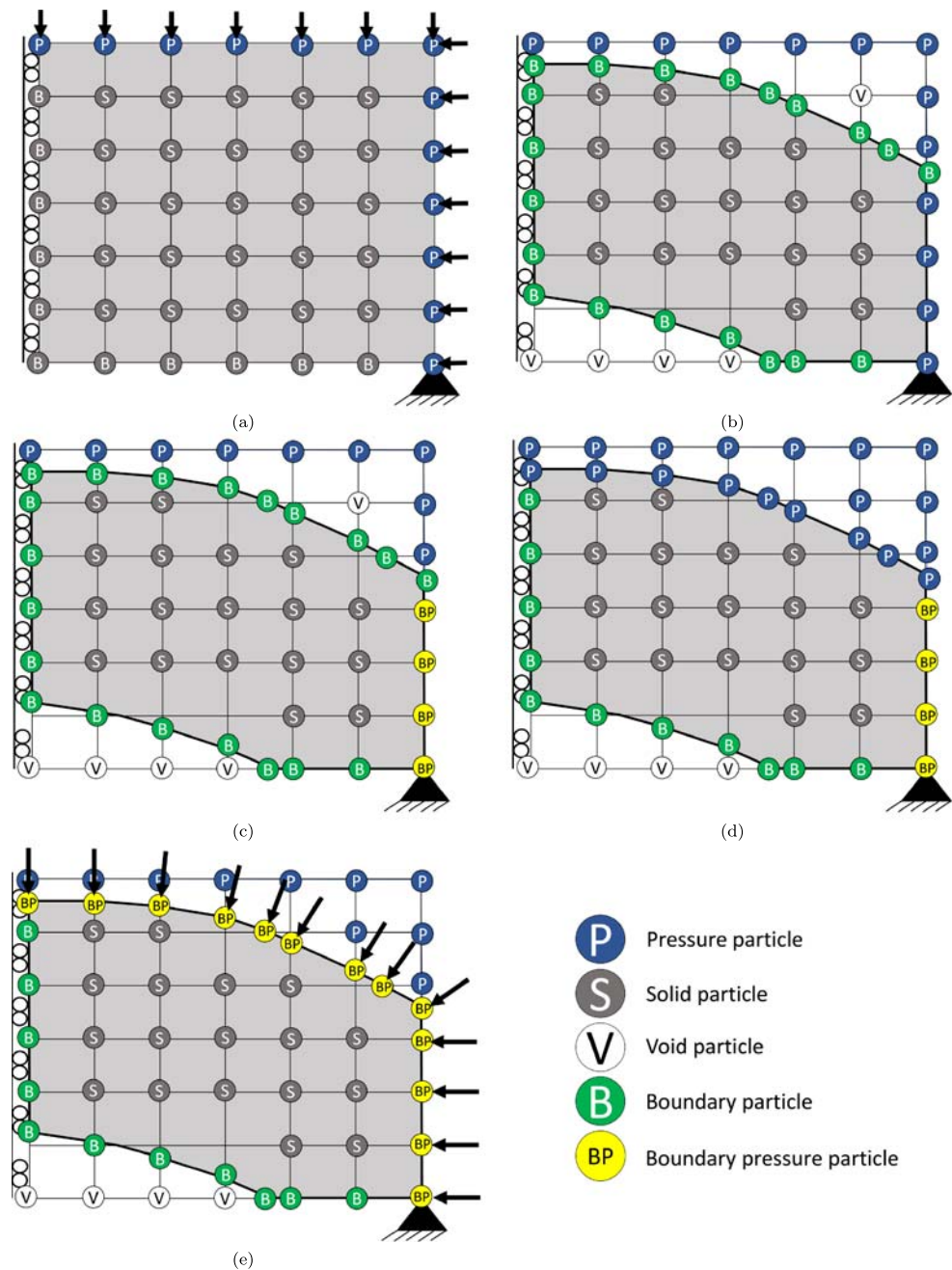
- solid particles that lie inside the structure,
- void particles that lie outside the structure,
- boundary particles that lie on the structural boundary,
- boundary pressure particles that indicate where pressure load should be applied.

Figure 3 illustrates the algorithm of moving boundaries. Figure 3a shows the initial arrangement. The particles in blue with symbol “P” indicate “pressure” particles, and the gray particles with “S” indicate solid particles. When the boundaries are updated (Fig. 3b), the signed distance,  $\phi$ , at the particles is calculated to classify them into solids (“S”) with  $\phi > 0$ , voids (“V”) with  $\phi < 0$ , and boundary (“B”) particles with  $\phi = 0$ . The initial pressure particles remain unchanged throughout optimization except for those initial pressure particles that also lie on the boundary. These become boundary pressure particles in subsequent iterations (“BP”) as shown in Fig. 3c. The reason for changing these boundary particles into “boundary pressure” instead of “pressure” is to stop the pressure load from advancing to the pressure-free portion of the boundary as will become apparent shortly. Also note that boundary pressure particles only appear after the first iteration; this is why they do not yet exist in Fig. 3a. In Fig. 3d, void or boundary particles that have pressure neighbors are transformed into pressure particles and this advances the pressure region. Neighboring between grid nodes and boundary particles is

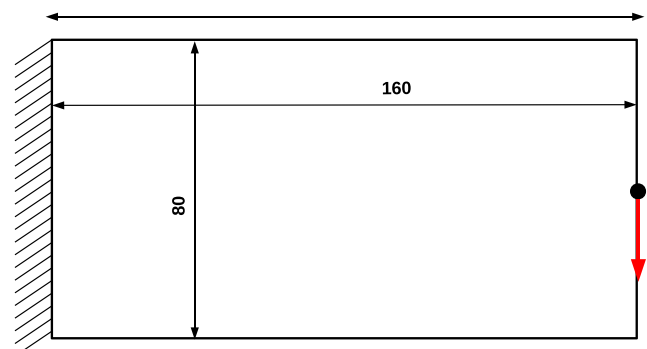
**Fig. 2** a Initial particle distribution. b Particle distribution after an optimization iteration



**Fig. 3** **a** Initial particle arrangement. **b** Boundary evolves and particle type assigned based on particle signed distance value. **c** Initial boundary pressure particles are identified. **d** Pressure region advances. **e** Boundary pressure particles are identified and pressure load is applied on boundary segments including these particles



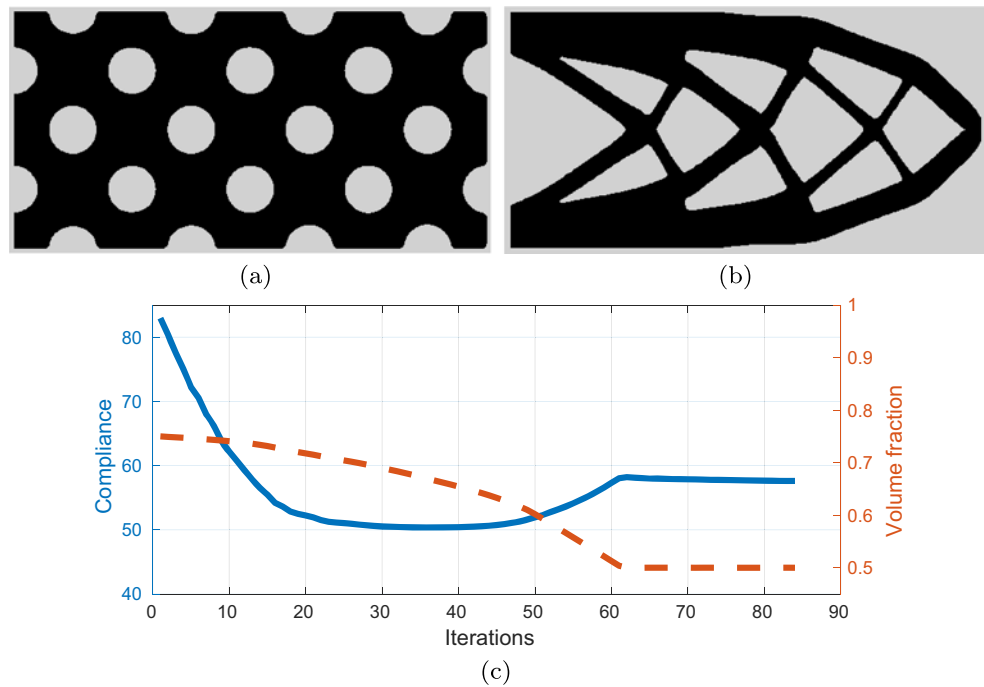
**Fig. 4** Elements cut twice by the boundary. **a** Possible configuration. **b** Not possible configuration



**Fig. 5** Cantilever beam example scheme

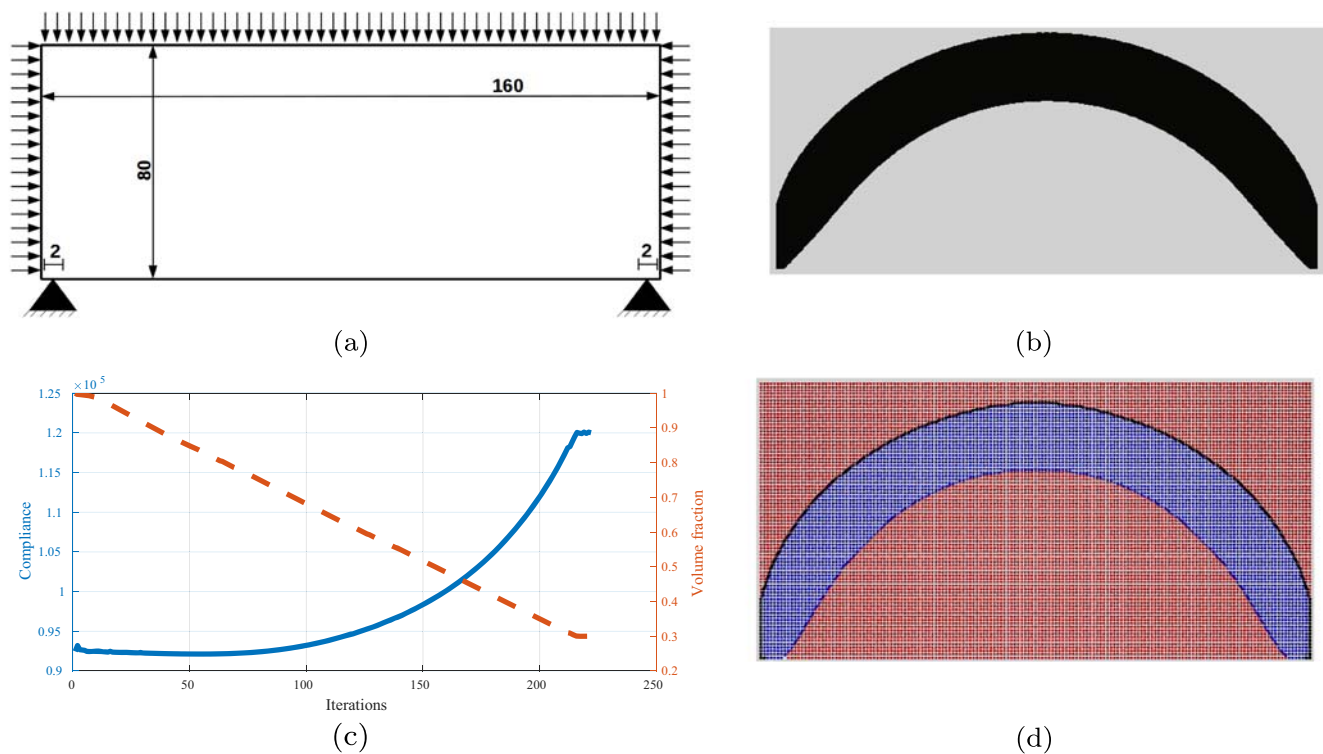


**Fig. 6** **a** Initial design configuration. **b** Optimum solution. **c** Convergence history and volume fraction plot for the cantilever beam



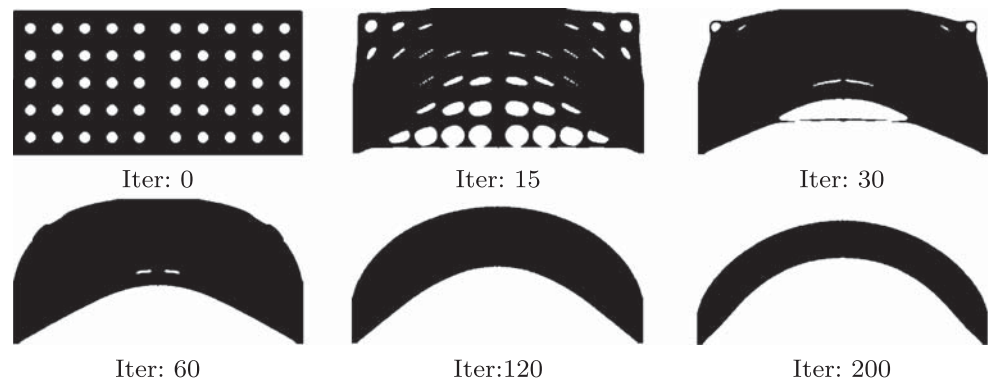
known by their connectivity on the level set mesh. The process is continued until the pressure region comes into contact with the boundary. Once the “P” type encounters a boundary particle, the particle is turned into “BP” type as shown in Fig. 3e. Had these particles remained as “pressure” particles, then all the “B” particles would eventually turn into “P,” thus

transferring the load also in the portion of the boundary that must remain pressure-free (left and bottom boundary particles). Thus, the separate “BP” type is used to stop further advancement of the pressure region on the load-free part of the boundary. Finally, the pressure load is applied to the part of the boundary on which the “BP” particles lie.



**Fig. 7** Arch structure. **a** Problem definition (initial domain with no holes). **b** Optimum solution. **c** Convergence history. **d** Particle distribution: void particles in red, solid particles in blue, pressure boundary particles in black

**Fig. 8** Snapshots of optimization history in the arch example



An important point is that in this work, the same mesh used for discretizing the level set function is also used as the background mesh for the domain integration in RKPM. For the particular case where an element is cut twice by a boundary, the following two configurations may arise:

The configuration shown in Fig. 4b is not possible due to the linear interpolation of the level set function. The configuration shown in Fig. 4a is possible, however, and the scheme for identifying the pressure boundary based on particle types is able to correctly apply the pressure loads for this instance because the load is transferred from point to point rather than from element to element. Thus, the load transfer stops once the pressure region encounters the first of the two segments within the element and does not proceed to the second segment.

We define in this problem that the outside pressure cannot be transferred to holes inside the structure. However, there is nothing fundamentally limiting in the methodology to consider problems with inner pressure like pressure vessels.

### 4.3 Boundary sensitivities

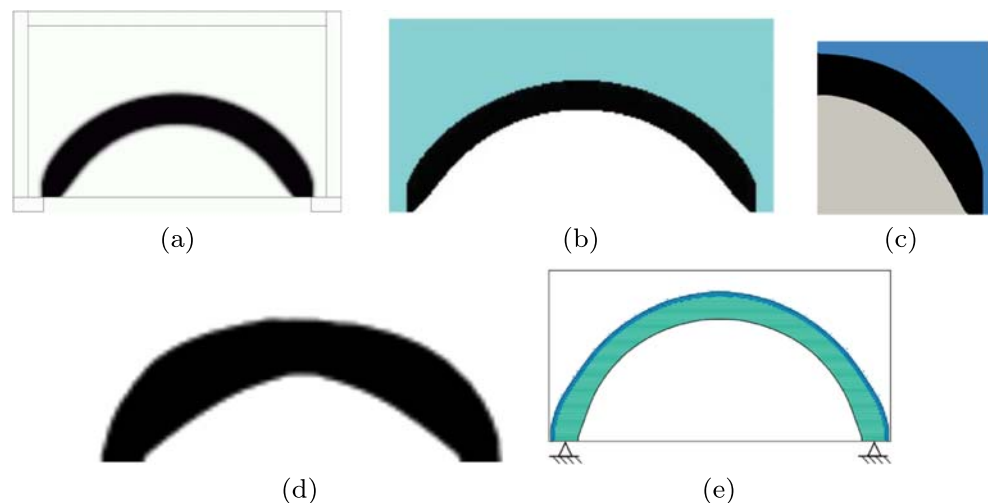
Shape sensitivities for the structural compliance function when the surface load is a pressure load oriented in the direction of the normal vector were derived by Allaire et al. (2004)

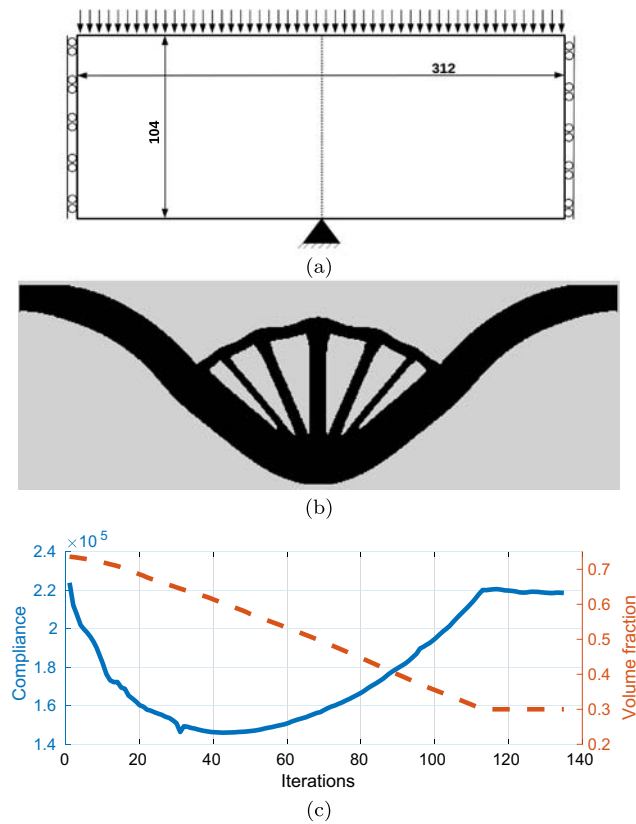
as

$$\frac{\partial f}{\partial x} = \int_{\Gamma_s} [-2\text{div}(p_0 u) - \sigma(u) \cdot \epsilon(u)] V_n d\Gamma_s, \quad (36)$$

where  $p_0$  is the pressure load,  $u$  is the displacement,  $\sigma(u) = D : \epsilon(u)$  is the stress tensor,  $\epsilon(u)$  is the strain tensor, and  $\Gamma_s$  represents the structural boundary. For the boundary points not on the pressure surface, the divergence term becomes zero. As points on the boundary are covered by the support domains of particles from both the solid and void regions, Young's modulus is interpolated using Eqs. (2) and (31) given in the “Reproducing kernel particle method” section. As can be seen in Eq. (36), the shape sensitivity includes stress and strain terms. The higher order continuity in the approximation of RKPM is another advantage over finite element analysis with linear elements where stress is discontinuous across the element edges. The RK approximation has the advantage of employing higher order smoothness with arbitrary order consistency, which avoids stress discontinuity in FEM with linear elements. Thus, smooth stresses can be obtained directly at the boundary points without any additional treatment such as stress recovery.

**Fig. 9** Arch solutions from the literature. **a** Sigmund and Clausen (2007) (SIMP). **b** Picelli et al. (2015) (BESO). **c** Picelli et al. (2019) (LSM). **d** Xia et al. (2015) (LSM). **e** Emmendoerfer et al. (2018) (LSM)



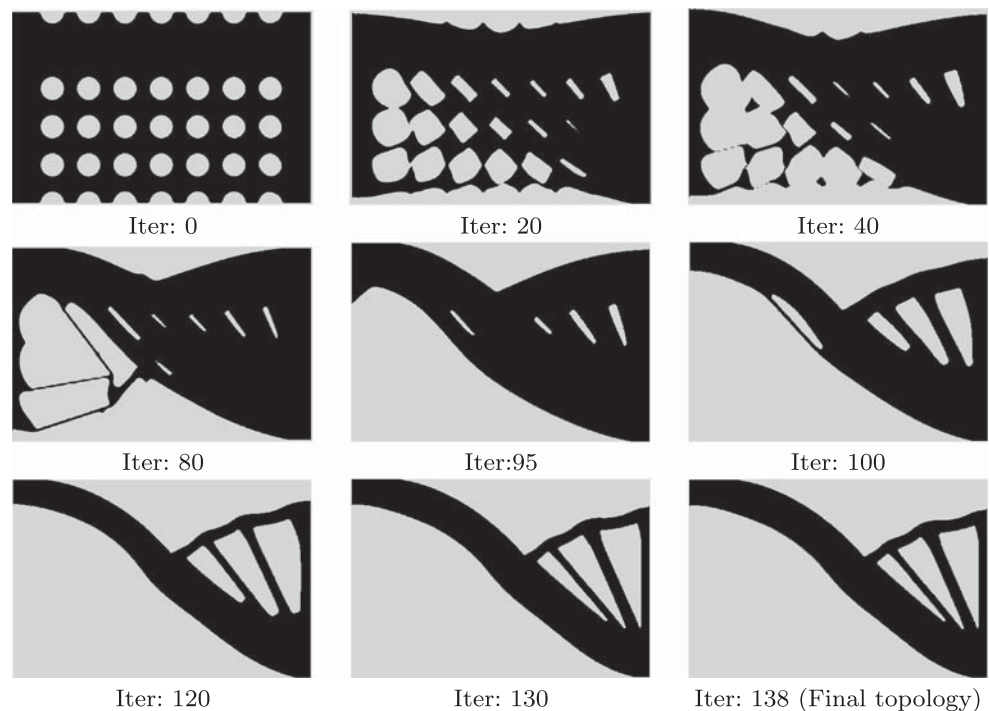


**Fig. 10** **a** Piston head structure problem definition. **b** Optimum solution. **c** Convergence history

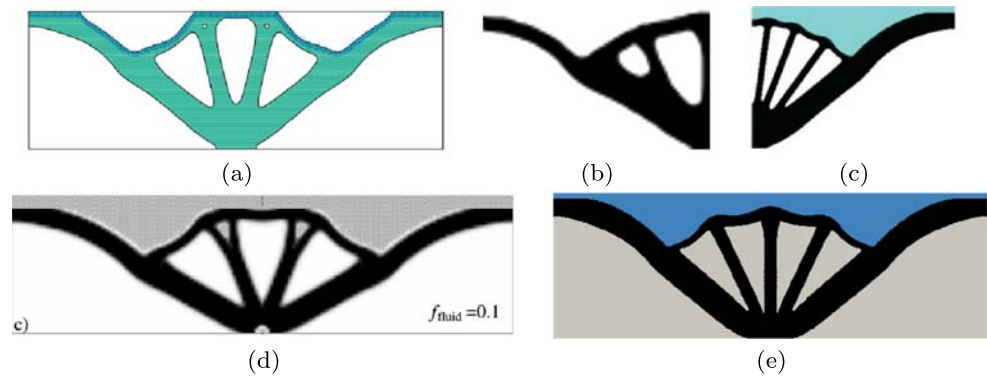
## 5 Numerical examples

This section starts with the cantilever beam with a fixed load commonly seen in many works in topology optimization. The aim is to validate the proposed method against this common benchmarking example before considering design-dependent pressure loads. As can be seen, the solution agrees with the well-known solutions in the literature (see, for example, Allaire et al. 2004). The following three benchmark examples for design-dependent pressure loading problems are investigated and compared with the literature. For all examples, a plane stress condition is assumed. For the solid material, Young's modulus is equal to 1, whereas for the void region, Young's modulus is set to be  $10^{-4}$ . A Poisson's ratio of 0.3 is used. For the domain integration,  $4 \times 4$  Gauss points are used in each background cell. A constant normalized support size of 1.5 times the size of a background cell was used for all particles as this was found to be the minimum support size that yields good convergence rate. This constant support size worked well even with the addition of new boundary particles at every iteration. For support sizes beyond the selected size and up to 3.5 times the size of a background cell, there was no change in the final solution. For each example, a number of initial hole configurations were investigated and obtained consistent solutions. Convergence of the objective is checked during 5 consecutive iterations and the tolerance is 0.001.

**Fig. 11** Snapshots of optimization history in the piston example



**Fig. 12** Piston solutions from the literature. **a** Emmendoerfer et al. (2018) (LSM). **b** Xia et al. (2015) (LSM). **c** Picelli et al. (2015) (BESO). **d** Sigmund and Clausen (2007) (SIMP). **e** Picelli et al. (2019) (LSM)



## 5.1 Cantilever beam

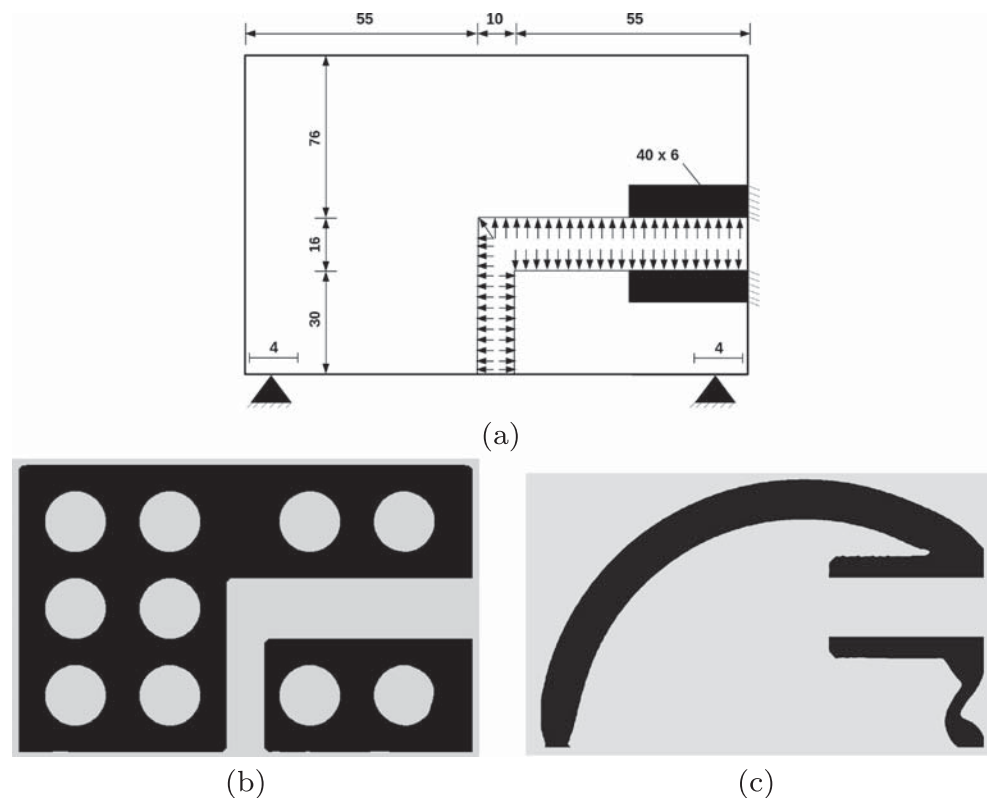
A cantilever beam subjected to a point load is shown in Fig. 5. For the background mesh, the design domain is discretized by  $160 \times 80$  cells. The meshfree particles are placed at the nodal locations of the background mesh, with new particles added at the new boundaries at every iteration as described in the “Level set topology optimization method” section. The mean compliance is minimized under a volume constraint of 50%. The initial and final topologies are shown in Fig. 6a and b, respectively. It can be seen in Fig. 6b that the proposed optimization methodology with the boundary point addition scheme obtains the familiar optimum topology. Figure 6c shows a smooth and stable convergence behavior similar to

what has been observed in the literature (see, for example, Allaire et al. 2004).

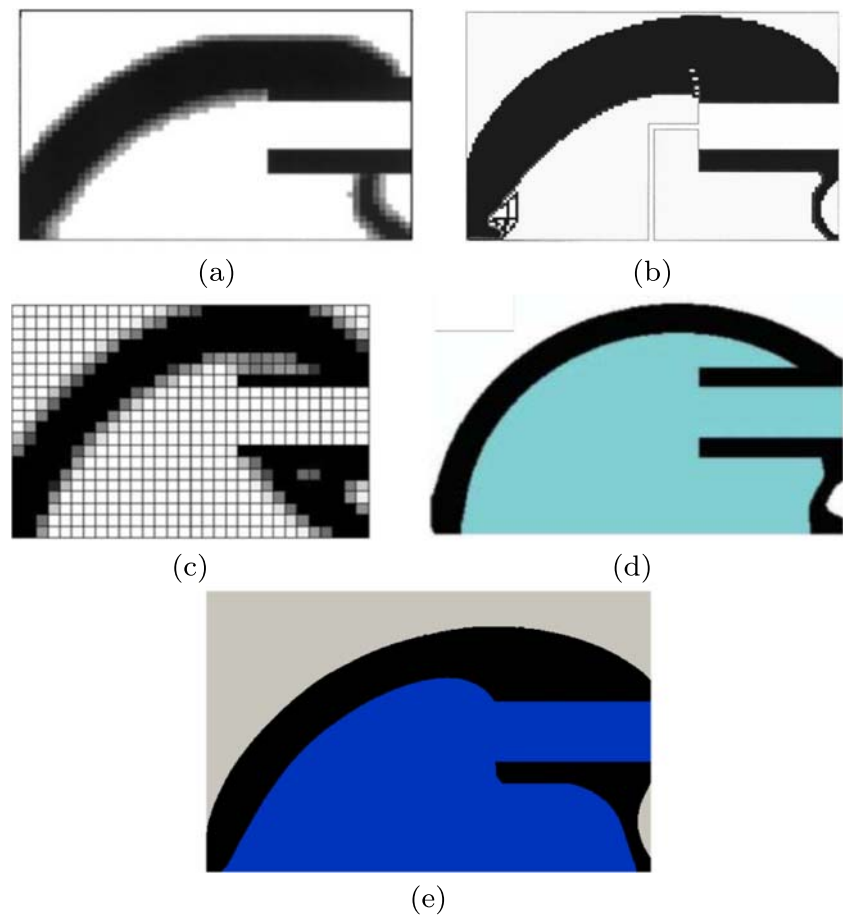
## 5.2 Arch structure

The arch example is a popular example used to validate topology optimization with design-dependent pressure loads. The model considered here is shown in Fig. 7a. A background mesh consisting of  $160 \times 80$  rectangular cells is used. The final volume fraction is set to 30%. The structure is subjected to a constant pressure load  $p = 1$  on the top, left, and right edges. The arch-like optimum solution in Fig. 7b agrees well with those obtained by previous works that considered this example (Sigmund and Clausen 2007;

**Fig. 13** Pressure chamber. **a** Problem definition. **b** Initial design with holes. **c** Optimum solution



**Fig. 14** Chamber solutions from the literature. **a** Chen and Kikuchi (2001) (SIMP). **b** Hammer and Olhoff (2000) (SIMP). **c** Zhang et al. (2008) (SIMP). **d** Picelli et al. (2015) (BESO). **e** Picelli et al. (2015) (LSM)



Picelli et al. 2015; Picelli et al. 2019; Xia et al. 2015; Emmendoerfer et al. 2018). These solutions are shown in Fig. 9 for comparison. This solution is also what was expected intuitively since spherical shapes are theoretically the ideal structures for pressure vessels. The convergence and volume fraction plots are given in Fig. 7 c. Similar to the cantilever beam of Fig. 6 c, the convergence is smooth and the rate is considered reasonable. The behavior of the compliance curve depends on the initial solution of the problem. The final topology and convergence history shown in Fig. 7 b and c, respectively, resulted from an initial configuration without any holes, i.e., infeasible solution. In these cases, the convergence is expected to move up because as the volume is reduced, the structure becomes more compliant (less stiff). A similar behavior can be seen in the literature where the authors have solved this example with different methods (see, for example, Picelli et al. 2015). Figure 8 shows snapshots of the optimization history for a different initial configuration with holes. As can be seen from these results, the consistent solution shown in Fig. 7b is obtained with and without holes in the initial design. Figure 7d illustrates the particle distribution in the solid (blue particles) and void (red particles) regions, with the pressure interface indicated by the black particles.

### 5.3 Piston head model

Another commonly solved example in the literature is the piston head example as shown in Fig. 10a. The authors have considered this example using SIMP by Sigmund and Clausen (2007) and Lee and Martins (2012), BESO by Picelli et al. (2015), and the level set method by Xia et al. (2015), Emmendoerfer et al. (2018), and Picelli et al. (2019). Roller boundary condition is applied on the sides and the center point is fixed (Fig. 9). Due to symmetry, only the right half of the model is solved here using  $156 \times 104$  background cells. A volume constraint of 30% is applied. The mirrored optimum topology is shown in Fig. 10b, whereas Fig. 10c shows the convergence history and volume fraction plots. Figure 11 illustrates the iterative process for a particular initial hole distribution. Figure 12 shows solutions obtained for the piston head example by other authors using different methodologies. Since different methodologies were used for each of these solutions with different specifications for the analysis such as different mesh sizes, the numbers are not directly comparable. We thus compare the solutions qualitatively based on the features they have in common. As can be seen, the arch-like curves near the lateral left and right walls and the elongated triangular holes in the center section are features



much similar to that obtained by other authors who previously solved this example. The solution obtained by RKPM in Fig. 10b is much similar with the ones obtained by Picelli et al. (2019) in which the level set method with equivalent nodal loads is used, and Picelli et al. (2015) using the BESO method. The main difference between the different approaches is in the number of internal structural members appearing. There is also a noticeable difference compared with the other level set methods by Emmendoerfer et al. (2018) and Xia et al. (2015). Although the overall looks are similar, the shapes of the holes are different in these two works and also the position of the top structural members is higher in Emmendoerfer et al. (2018).

#### 5.4 Pressure chamber model

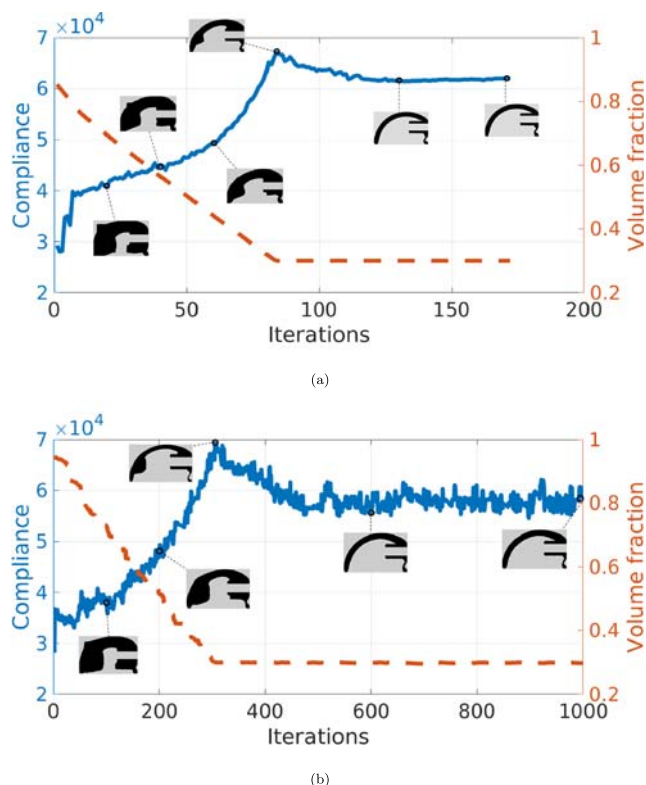
The pressurized chamber example was first proposed by Hammer and Olhoff (2000) and Chen and Kikuchi (2001). The only subsequent works that considered this problem was Zhang et al. (2008) using SIMP method and Picelli et al. (2015) using BESO method. The problem definition as shown in Fig. 13a is solved with  $120 \times 76$  rectangular background cells. The two  $40 \times 60$  flanges shown in black in Fig. 13a are considered fixed and are excluded from the design domain. The applied pressure  $p = 1$  is indicated by the arrows. Starting from an initial design with no holes, the final solution

in Fig. 13b is obtained. The same solution is also obtained from a design with a different initial configuration such as the one shown in Fig. 13, which shows the independency of the results on the initial design. The optimized structure agrees well with the ones obtained in the previous literature as shown in Fig. 14, especially the one obtained by Picelli et al. (2015). An interesting point here is that in this work, well-defined structural boundaries can be obtained with a relative small grid. For example, the similar solution to the present work obtained by Picelli et al. (2015) in Fig. 14 (d) used 57,000 finite elements, whereas here, we only use 9120 level set elements. The SIMP results obtained by Chen and Kikuchi (2001) and Hammer and Olhoff (2000) show differences with the RKPM solution in Fig. 13c. The top structural member appears to be thicker with solid covering all the area on top of the horizontal fluid region. The support at the right-hand side corner is also thinner in these examples. These differences are possibly due to the slightly different shape of the pressure region used in these examples as shown in Fig. 14b. Similar differences can also be seen compared with the optimum structure by Picelli et al. (2019). In this work, the level set method was used with ersatz material approximation and the loads were transformed into work equivalent nodal loads.

The geometry and changing loading direction make this example more challenging compared with the arch and piston examples. Based on the literature, one of the most common ways to deal with design-dependent loads using a fixed grid is to transform them into equivalent nodal loads (for example, in Hammer and Olhoff (2000), Du and Olhoff (2004), and Lee and Martins (2012) using SIMP, and Emmendoerfer et al. (2018) and Picelli et al. (2019) using the level set method). Thus, we compare the performance of the RKPM approach with an equivalent finite element method with work-equivalent nodal loads for this example. The implementation for the specific fixed grid approach we used here can be found in Neofytou et al. (2019). The same level set algorithm is used for the two approaches, whereas for the analysis, we simply replaced RKPM with the fixed grid FEA for a fair comparison. Also, no regularization has been used in either approach. The convergence history and volume fraction plots for the two methodologies are shown in Fig. 15. As can be seen, although the final topologies are very similar, RKPM has a much smoother and more stable behavior compared with the FEA approach. This results in a much faster convergence, with RKPM converging at 171 iterations, whereas the fixed grid FEA struggles to converge even after 1000 iterations have passed.

## 6 Conclusions

This paper presented a new level set topology optimization method with RKPM for compliance minimization of



**Fig. 15** Convergence history comparison between **a** RKPM and **b** fixed grid FEA with work-equivalent nodal loads for the chamber example

structures subjected to design-dependent pressure loads. Design-dependent problems in topology optimization are challenging because of the necessity to identify the surface for the loads to act and apply the pressure at the new boundaries. To address this challenge, the reproducing kernel particle method was used along with level set topology optimization. The combination of the two methods for the specific case of design-dependent loads in topology optimization is the main contribution of this work. The level set method provides the clear boundary representation and RKPM offers the freedom to place particles anywhere in the design domain and apply the pressure at the new boundaries. As shown through the numerical examples, the proposed methodology gives results that are in good agreement with the results obtained by different methods in the literature. This is the first method to use a meshfree approach to handle hydrostatic pressure loads in the topology optimization framework. The simplicity with which the meshfree method can handle the clear boundaries and the design-dependent loading by directly applying the pressure loads on the relevant structural boundaries without the need of any special treatments or remeshing demonstrates an advantage of this methodology. This can be advantageous when using separate governing equations, as we have the explicitly defined boundaries and can put particles directly on the boundary. In comparison with discrete (BESO or TOBS) methods, we do not have jagged surfaces and we model the correct smooth boundaries. In comparison with past LSM, we do not need to interpolate the pressure loads at the boundaries as we put the pressure particles wherever convenient. These two features can be particularly advantageous when dealing with more complex problems, like FSI, where the discretization at the boundaries is very important for the approximation and convergence of the analysis. In addition, RKPM offers higher order approximation of the equations. The RK approximation has the advantage of employing higher order smoothness with an arbitrary order consistency. The RK “smooth approximation” allows for Young’s modulus interpolation to yield smooth Young’s modulus distribution for better conditioned boundary sensitivities. This leads to a faster converge rate of the optimization procedure. Before tackling more complex problems, herein we show that our method addresses correctly the hydrostatic problem, which is the base for any design-dependent loading problem we aim to address in the future.

**Acknowledgments** The authors would like to thank the Numerical Analysis Group at the Rutherford Appleton Laboratory for their FORTRAN HSL packages (HSL, a collection of Fortran codes for large-scale scientific computation. See <http://www.hsl.rl.ac.uk/>). Furthermore, we would like to thank professor Jiun-Shyan Chen’s group at the University of California San Diego (UCSD) for providing their expertise on the reproducing kernel particle method. For a 2D RKPM open source code developed by the group, the reader is referred to Huang et al. (2019).

**Funding information** We thank the support of the Engineering and Physical Sciences Research Council, fellowship grant EP/M002322/2.

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

**Replication of results** Information on the data underpinning the results presented here, including how to access them, can be found in the Cardiff University data catalogue at <http://doi.org/....> The data from these numerical investigations are available by contacting the authors.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Allaire G, Jouve F, Toader AM (2004) Structural optimization using sensitivity analysis and a level-set method. *J Comput Phys* 194: 363–393
- Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. *Int J Numer Methods Eng* 37:229–256
- Bourdin B, Chambolle A (2003) Design-dependent loads in topology optimization. *Control, Optimisation and Calculus of Variations* 9: 19–48
- Chen BC, Kikuchi N (2001) Topology optimization with design-dependent loads. *Finite Elem Anal Des* 37:57–70
- Chen JS, Pan C, Wu CT, Liu WK (1996) Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Comput Methods Appl Mech Eng* 139(1–4):195–227
- Chen JS, Pan C, Wu CT (1997) Large deformation analysis of rubber based on a reproducing kernel particle method. *Comput Mech* 19(3): 211–227
- Chen JS, Wu CT, Yoon S, You Y (2001) A stabilized conforming nodal integration for Galerkin mesh-free methods. *Int J Numer Methods Eng* 50:435–466
- Chen J, Basava RR, Zhang Y, Csapo R, Malis V, Sinha U, Hodgson J, Sinha S (2017a) Pixel-based meshfree model of skeletal muscles. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization* 4(2):73–85
- Chen JS, Hillman M, Chi SW (2017b) Meshfree methods: progress made after 20 years. *J Eng Mech* 143(4)
- Cho S, Kwak J (2006) Topology design optimization of geometrically non-linear structures using meshfree method. *Comput Methods Appl Mech Eng* 195:5909–5925
- Dolbow J, Belytschko T (1999) Numerical integration of the Galerkin weak form in meshfree methods. *Comput Mech* 23:219–230
- Du J, Olhoff N (2004) Topological optimization of continuum structures with design-dependent surface loading - part i: new computational approach for 2d problems. *Struct Multidiscip Optim* 27:151–165

- Du Y, Luo Z, Tian Q, Chen L (2009) Topology optimization for thermo-mechanical compliant actuators using mesh-free methods. *Eng Optim* 41(8):753–772
- Emmendoerfer H, Fancello EA, Silva ECN (2018) Level set topology optimization for design-dependent pressure load problems. *International Journal for Numerical Methods in Engineering* online
- Hammer VB, Olhoff N (2000) Topology optimization of continuum structures subjected to pressure loading. *Struct Multidiscip Optim* 19:85–92
- Hedges LO, Kim HA, Jack RL (2017) Stochastic level-set method for shape optimisation. *J Comput Phys* 348:82–107
- Huang TH, Wei H, Chen JS, Hillman M (2019) RKPM2D: an open-source implementation of nodally integrated reproducing kernel particle method for solving partial differential equations. *Computational Particle Mechanics*, <https://doi.org/10.1007/s40571-019-00272-x>
- Isakari H, Kondo T, Takahashi T, Matsumoto T (2017) A level-set-based topology optimisation for acoustic-elastic coupled problems with a fast BEM-FEM solver. *Comput Methods Appl Mech Eng* 315:501–521
- Jenkins N, Maute K (2016) An immersed boundary approach for shape and topology optimization of stationary fluid-structure interaction problems. *Structural and* 54:1191–1208
- Khan W, Islam S, Ullah B (2018) Structural optimization based on meshless element free Galerkin and level set methods. *Comput Methods Appl Mech Eng*
- Kim NH, Choi KK, Botkin ME (2003) Numerical method for shape optimization using meshfree method. *Struct Multidiscip Optim* 24: 418–429
- Lee E, Martins JRRA (2012) Structural topology optimization with design-dependent pressure loads. *Comput Methods Appl Mech Eng* 233–236:40–48
- Liu WK, Jun S, Zhang YF (1995) Reproducing kernel particle methods. *Int J Numer Methods Fluids* 20:1081–1106
- Liu WK, Jun S, Sihling DT, Chen Y, Hao W (1997) Multiresolution reproducing kernel particle method for computational fluid dynamics. *Int J Numer Methods Fluids* 24(12):1391–1415
- Luo Z, Zhang N, Gao W, Ma H (2012) Structural shape and topology optimization using a meshless Galerkin level set method. *Int J Numer Methods Eng* 90:369–389
- Neofytou A, Picelli R, Chen J, Kim HA (2019) Level set topology optimization for design dependent pressure loads: a comparison between FEM and RKPM. *AIAA:2019–3559*. <https://doi.org/10.2514/6.2019-3559>
- Picelli R, Vicente WM, Pavanella R (2015) Bi-directional evolutionary structural optimization for design-dependent fluid pressure loading problems. *Eng Optim* 47(10):1324–1342
- Picelli R, Townsend S, Brampton C, Norato J, Kim H (2018) Stress-based shape and topology optimization with the level set method. *Comput Methods Appl Mech Eng* 329:1–23
- Picelli R, Neofytou A, Kim HA (2019) Topology optimization for design-dependent hydrostatic pressure loading via the level set method. *Struct Multidisc Optim* 60:1313–1326
- Sandilya K, Du Z, Chung H, Kim HA, Jauregui C, Townsend S, Picelli R, Zhou XY, Hedges L (2018) OpenLSTO: open-source software for level set topology optimization. In: *Multidisciplinary analysis and optimization conference*, p 3882
- Sethian J (1996) A fast marching level set method for monotonically advancing fronts. *Proc Natl Acad Sci* 93(4):1591–1595
- Shu L, Wang MY, Ma Z (2014) Level set based topology optimization of vibrating structures for coupled acoustic-structural dynamics. *Comput Struct* 132:34–42
- Sigmund O, Clausen PM (2007) Topology optimization using a mixed formulation: an alternative way to solve pressure load problems. *Comput Methods Appl Mech Eng* 196:1874–1889
- Sivapuram R, Picelli R (2017) Topology optimization of binary structures using Integer Linear Programming. *Finite Elem Anal Des* 139:49–61
- Wang MY, Wang X, Guo D (2003) A level set method for structural topology optimization. *Comput Methods Appl Mech Eng* 192: 227–246
- Wang HP, Wu C, Chen J (2014) A reproducing kernel smooth contact formulation for metal forming simulations. *Comput Mech* 54(1): 151–169
- Xia Q, Wang MY, Shi T (2015) Topology optimization with pressure load through a level set method. *Comput Methods Appl Mech Eng* 283: 177–195
- You Y, Chen JS, Lu H (2003) Filters, reproducing kernel, and adaptive meshfree method. *Comput Mech* 31(3–4):316–326
- Zhang H, Zhang X, Liu S (2008) A new boundary search scheme for topology optimization of continuum structures with design-dependent loads. *Struct Multidisc Optim* 37:121–129
- Zhang Y, Ge W, Zhang Y, Zhao Z, Zhang J (2018) Topology optimization of hyperelastic structure based on a directly coupled finite element and element-free Galerkin method. *Adv Eng Softw* 123:25–37
- Zhou JX, Zou W (2008) Meshless approximation combined with implicit topology description for optimization of continua. *Struct Multidiscip Optim* 36:347–353

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.